High Level Architecture

Interface Specification Overview

Defense Modeling and Simulation Office
phone: (703) 998-0660 FAX: (703) 998-0667
http://www.dmso.mil

Outline

I. INTRODUCTION

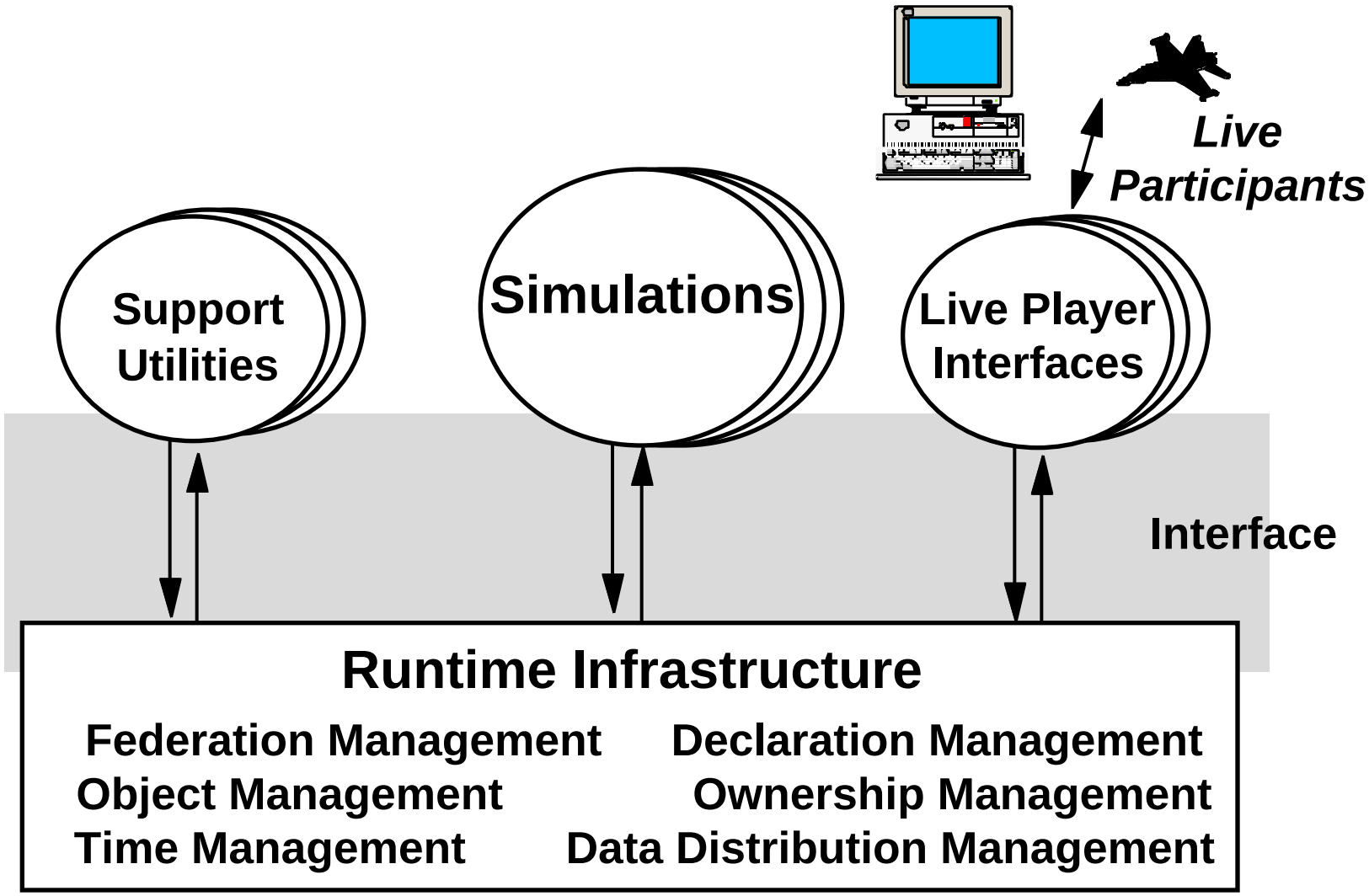
II. HLA I/F SPECIFICATION SERVICE GROUPS

III. USE OF THE INTERFACE IN A TYPICAL FEDERATION EXECUTION

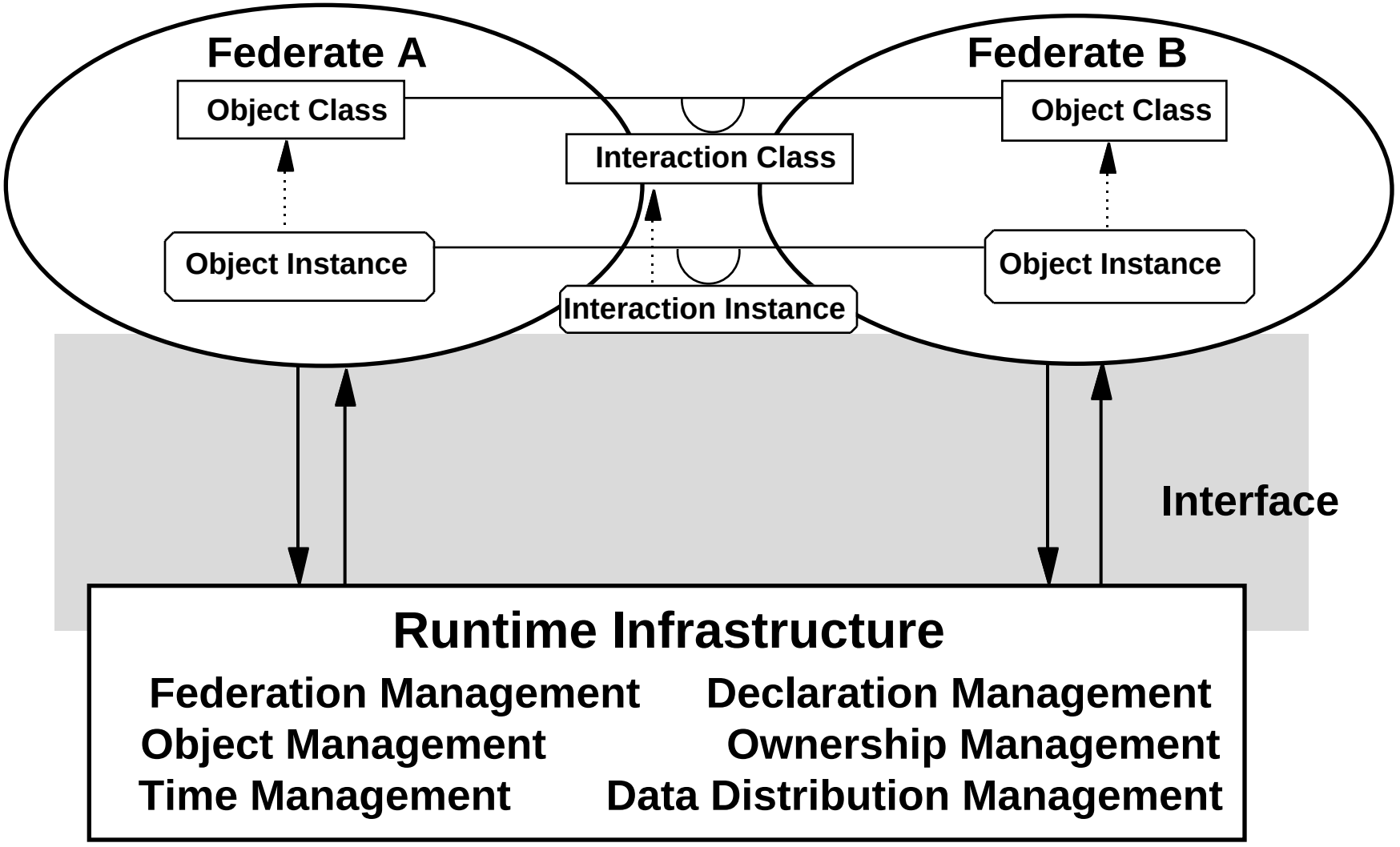
IV. SUPPORTING DOCUMENTS AND TECHNICAL PAPERS

V. CONCLUSION

Functional View of the Architecture



Logical View of the Architecture



Rationale for an Interface Specification

- **Provides a specification of the functional interfaces between federates and the RTI**
- **Facilitates, through a common, well defined, consistent set of interface definitions;**
  - **INTEROPERABILITY within simulations, among simulations of a federation, and across functional M&S communities**
  - **REUSE of simulation components**

HLA Interface Specification Design Goals and Strategies

HIGH QUALITY

- Completeness
- Consistency
- Conciseness

USABLE

- Power and User Discretion

UNDERSTANDABLE

- Intuitive reflection of entity roles
- Symmetry and organization of Services

AUTHORITATIVE

- Primacy of Specification

HLA Interface Specification Characteristics

- **Provides a specification of the functional interfaces between federates and the RTI**
  - **65 interfaces in six service groups**
- **Each service specification includes:**
  - **Name and Descriptive Text**
  - **Supplied Parameters**
  - **Returned Parameters**
  - **Pre-conditions**
  - **Post-conditions**
  - **Exceptions**
  - **Related Services**
- **Application Programmers Interface (API) in CORBA IDL, Ada '95 and C++**

Outline

I. INTRODUCTION

II. HLA I/F SPECIFICATION SERVICE GROUPS

III. USE OF THE INTERFACE IN A TYPICAL FEDERATION EXECUTION

IV. SUPPORTING DOCUMENTS AND TECHNICAL PAPERS

V. CONCLUSION

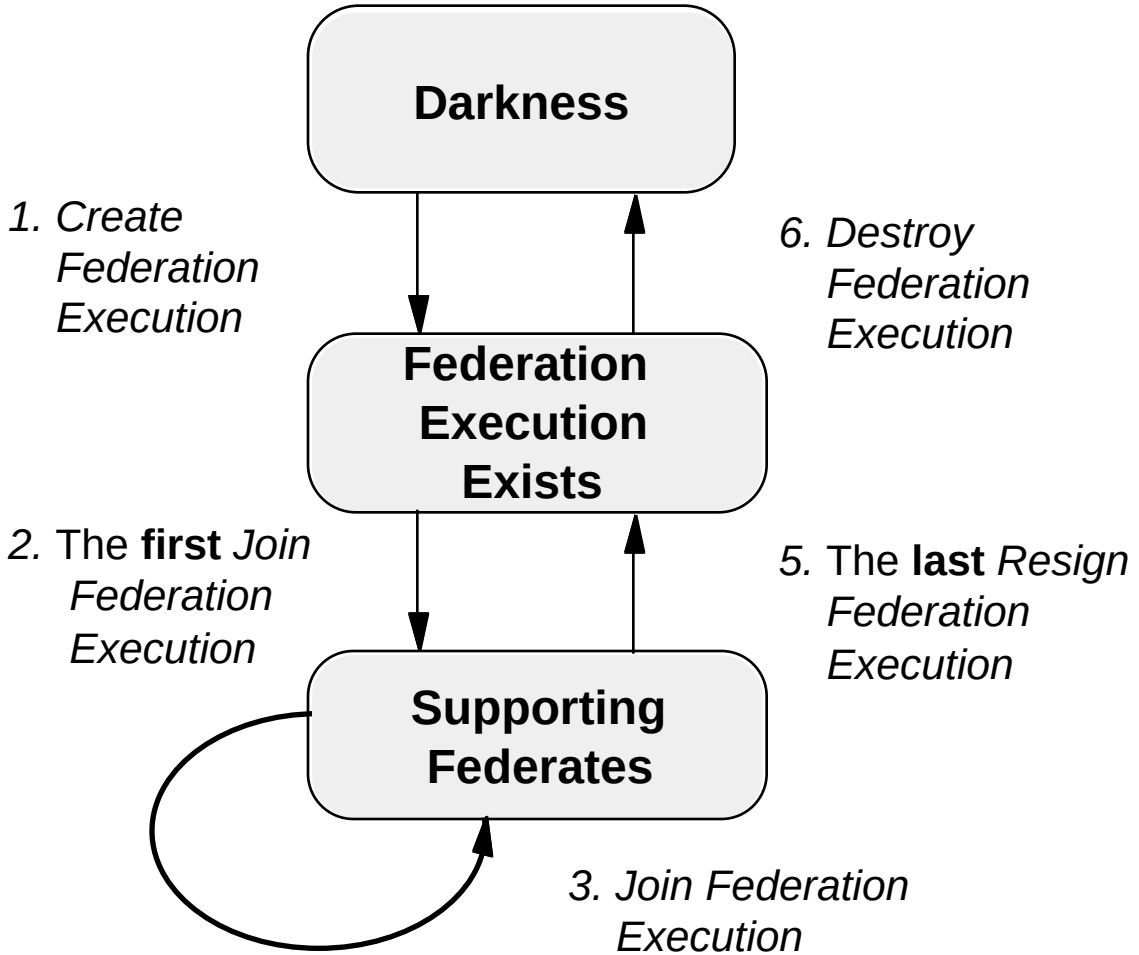
Six HLA Runtime Infrastructure Service Groups

- Federation Management (17)
- Declaration Management (6)
- Object Management (17)
- Ownership Management (9)
- Time Management (10)
- Data Distribution Management (6)

Federation Management

- **Coordinate federation-wide activities throughout the life of a federation execution**
  - **Used by federates to manage a federation execution to meet their needs**
  - **Includes RTI initialization data for initializing name space, transportation and ordering defaults, as well as routing space names and dimensions**
- **Interface functions include**
  - **Creation and destruction of a federation execution**
  - **Joining and resigning of a federate**
  - **Coordination of federation saves**
  - **Pausing and resuming a federation execution**

Federation Management



Declaration Management

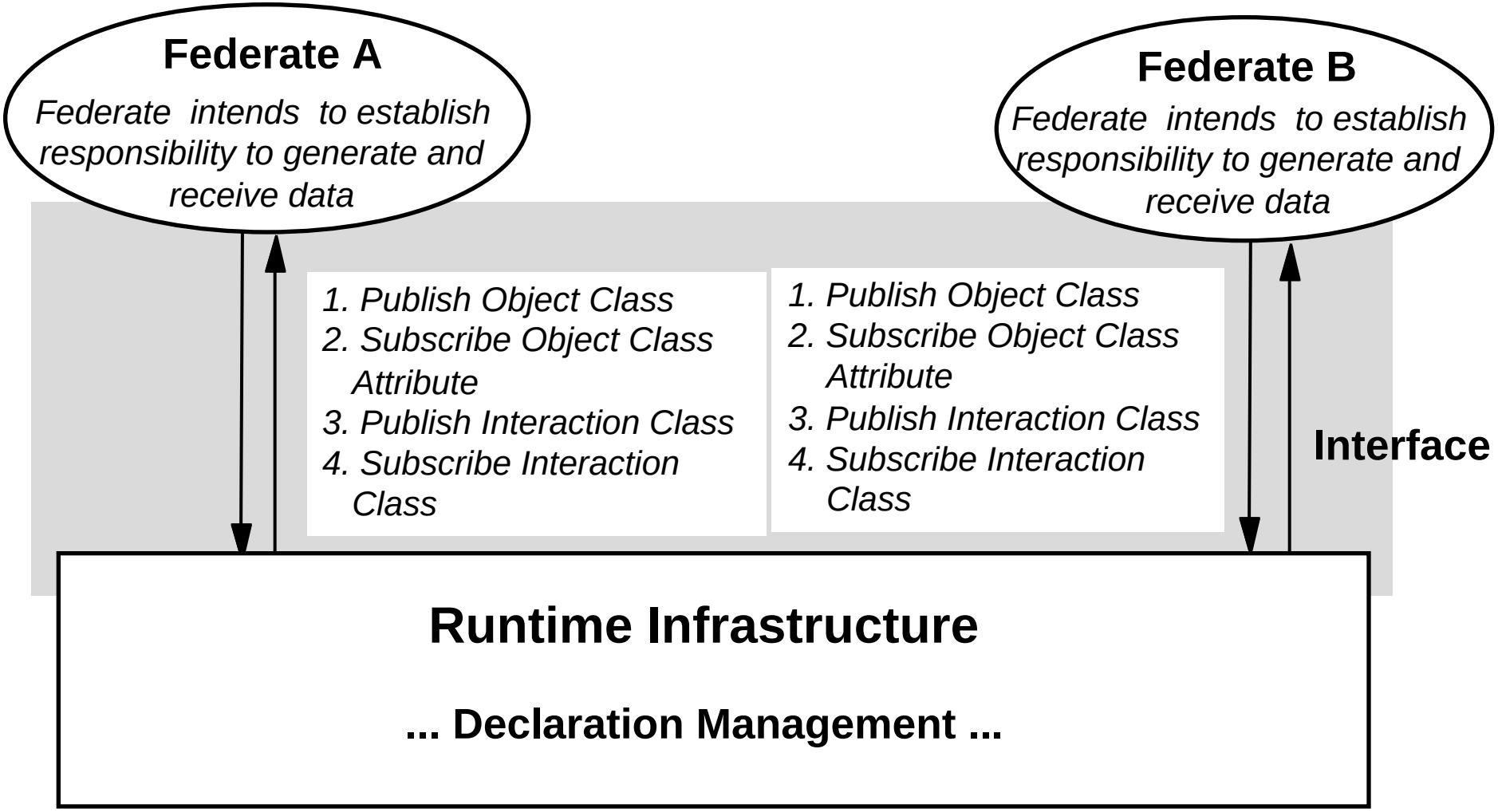
- **Allow federates to specify the types of data they will send or receive by object class and attribute name and by interaction class from the FOM**
- **Interface functions include:**
  - **Publish Object Class / Interaction Class**

    Object class attributes and interaction classes that the federate is able to update or send
  - **Subscribe Object Class Attribute / Interaction Class**

    Object classes and attributes and interaction classes that the federate desires to receive
  - **Control Updates / Interactions**

    Feedback to the federates from the RTI when attribute updates and interactions should be sent given an interest by other federates

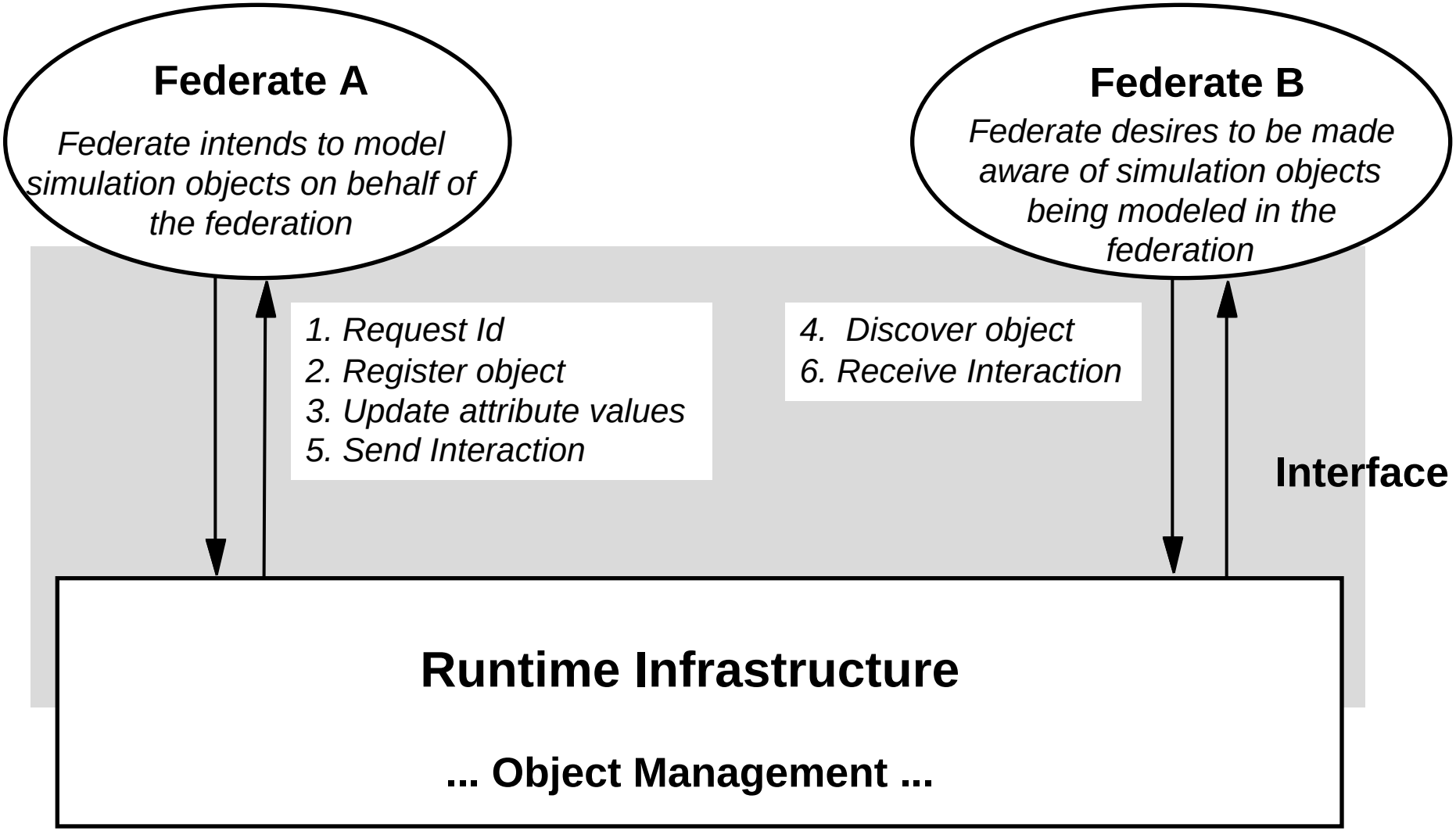
Declaration Management



Object Management

- **Supports creation, modification, and deletion of objects, their attributes and the interactions they produce**
- **Interface functions include:**
  - **Federate requests for IDs**
  - **Registering and discovering objects**
  - **Updating and reflecting object attributes**
  - **Sending and receiving interactions**
  - **Deleting and removing objects**
  - **Changing default transportation and event ordering types**

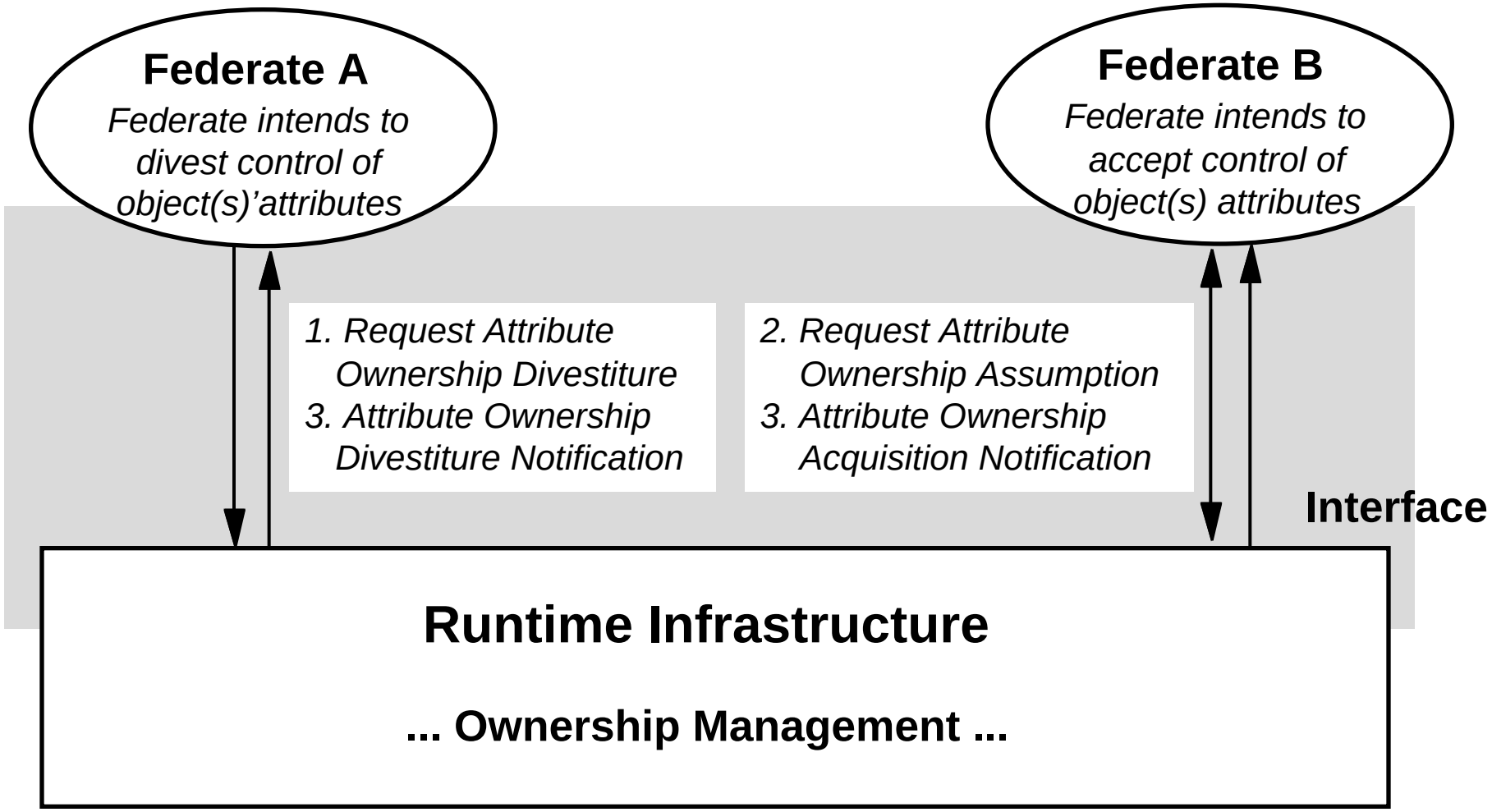
Object Management Example



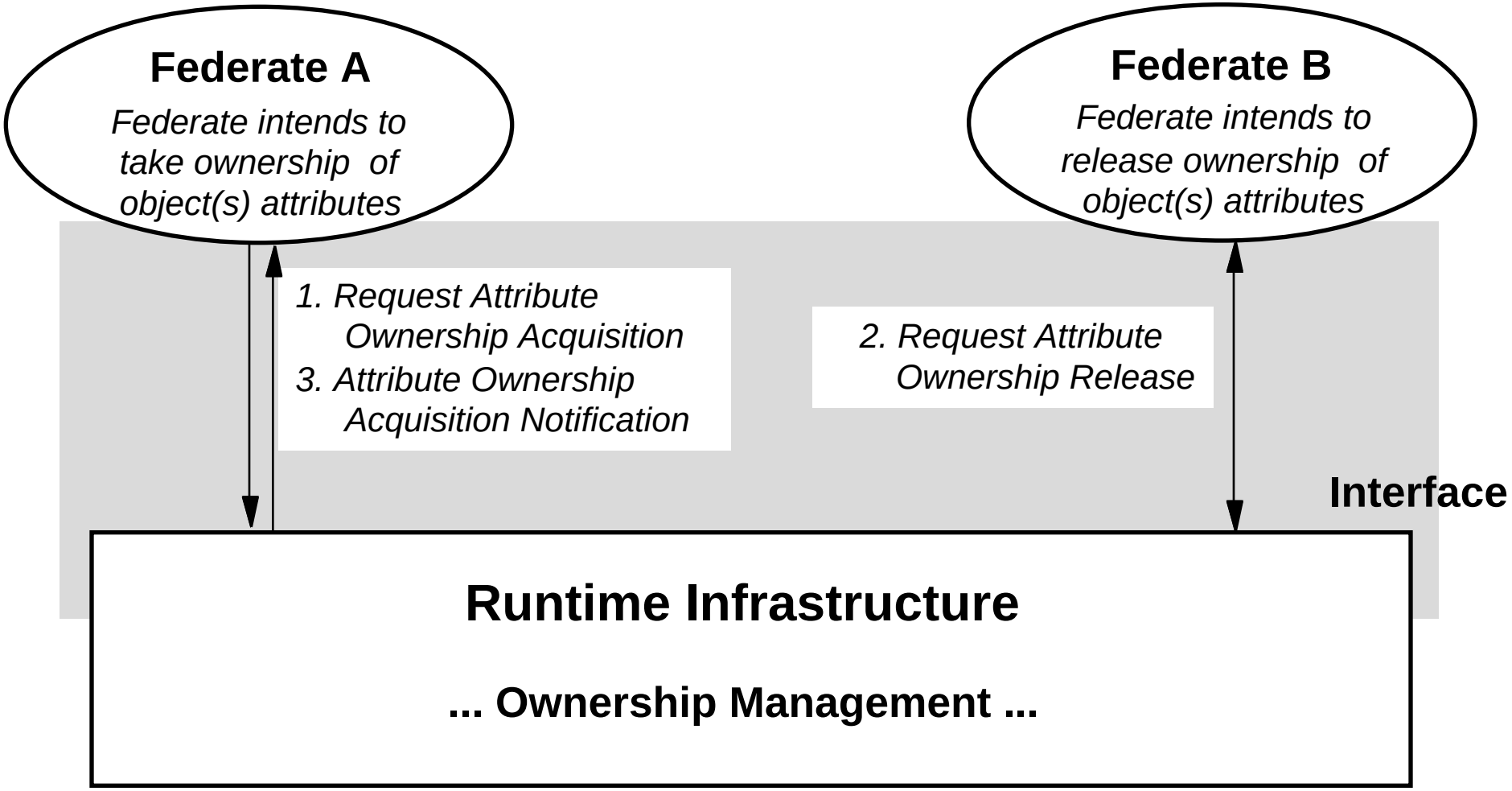
Ownership Management

- **Allow federates to transfer ownership of object attributes**
  - **Federates transfer ownership based on federation execution design plans and the RTI arbitrates transactions**
  - **Offers both ‘push’ or ‘pull’ based transactions**
  - **Acquisition requires current publication and subscription declarations for attribute**
- **Interface functions include:**
  - **Request ownership divestiture and assumption**
  - **Request ownership acquisition and release**
  - **Notification of divestiture and acquisition**
  - **Query attribute ownership**

Divesting Ownership Example



Requesting Ownership Example

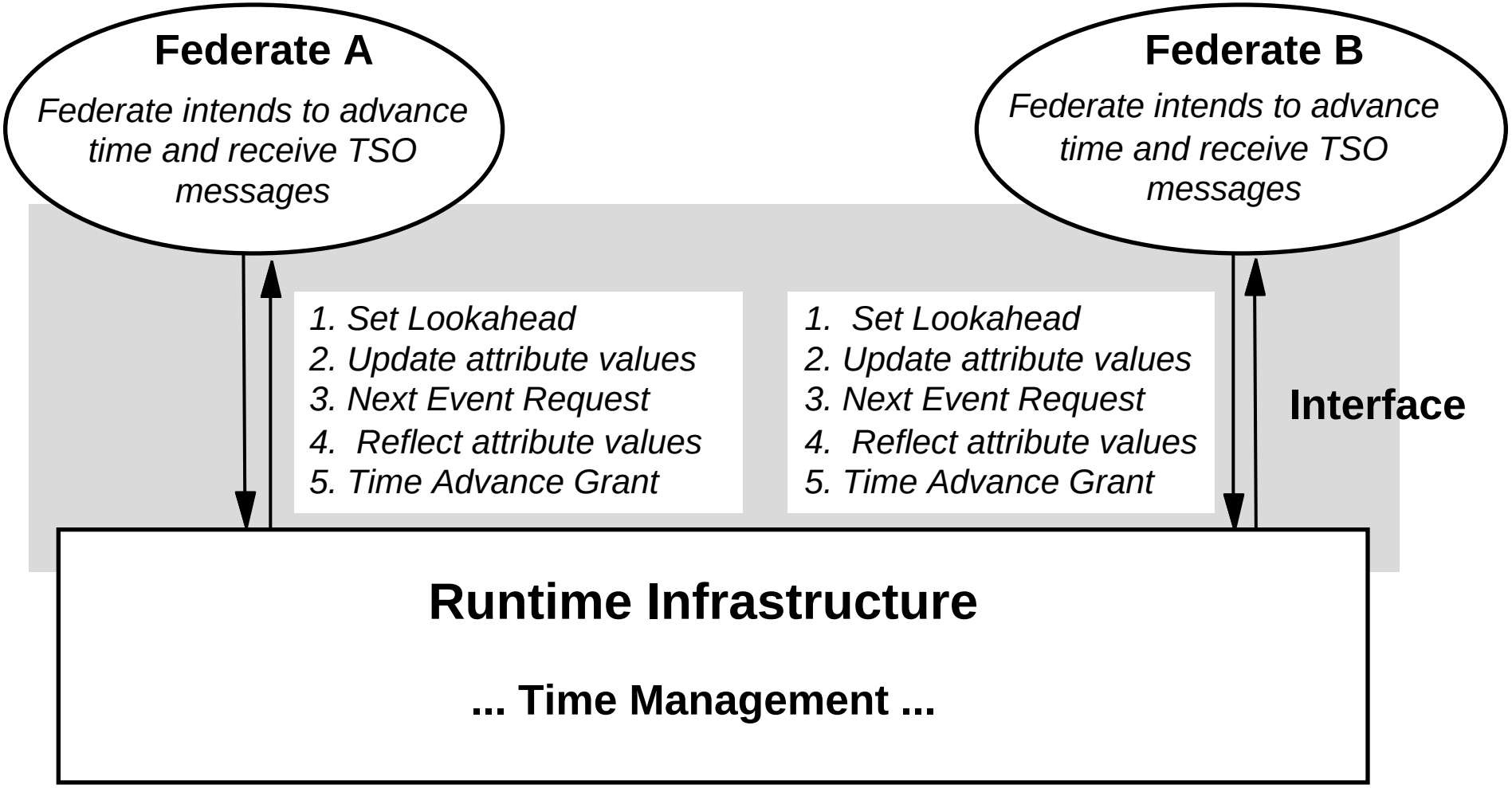


Time Management

- **Control advancement of federates along with federation time**
  - **Coordinated with object management services to support causal behavior across the federation**
  - **Designed to support federates with different ordering and delivery requirements**
- **Interface functions include**
  - **Request current values of time**

    Federation time, federate's logical time (LT), lower bound time stamp (LBTS), minimum next event time
  - **Set and request lookahead**
  - **Time advance request, next event and flush queue request, and grant**

Time Management Example

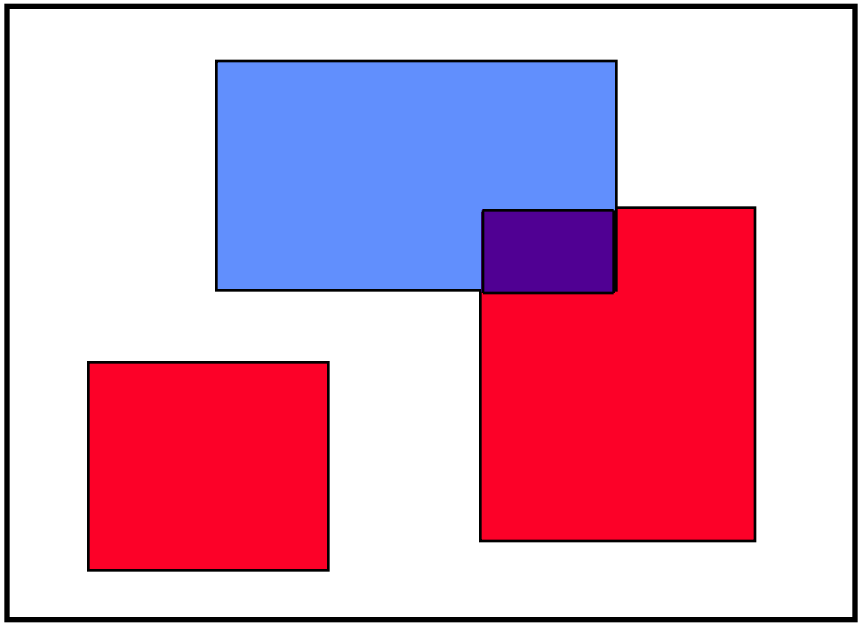


Data Distribution Management

- **Allow federates to specify the distribution conditions for the specific data they send or expect to receive**
  - **RTI uses this information to route data as specified in declaration management services**
  - **Not bound by FOM, data distribution can be managed based on other characteristics of objects important to particular federation execution**
  - **Federation design creates 'routing spaces' for use during runtime; these are specified at federation creation time**
- **Interface functions include**
  - **Create and modify 'update' and 'subscription' regions to bound routing space**
  - **Associate update regions with specific object instances**
  - **Notification to change thresholds for regions**

Illustration of DDM Services

Two Dimensional Interest Space



Update Region

Subscription Region

Overlap Region - Published Data Sent to Subscribing Federate

The Role of the Federate in DDM

- **Federates using DDM services associate objects with regions of interest**

Create Subscription Region [7.2]

Specify conditions under which they desire to receive updates and interactions

Create Update Region [7.1]

Specify conditions under which they agree to produce updates and interactions

Associate Update Region [7.3]

Associate an object with an update region

Modify Region or Associate Update Region [7.5, 7.3]

Adjustment to the bounds on the associated regions based on state of change of objects within the region

The Role of the RTI in DDM

- **The routing space, regions, and association data is used by the RTI to distribute data**
- **When an update region and subscription regions of different federates overlap data is routed**
  - **The RTI ensures that the attribute updates and interactions associated with that update region are routed to federates with subscription regions which overlap the sender's update region**

Change Thresholds [7.4]

The RTI provides feedback to federate on the amount of change in extent which will lead to data distribution changes

Outline

I. INTRODUCTION

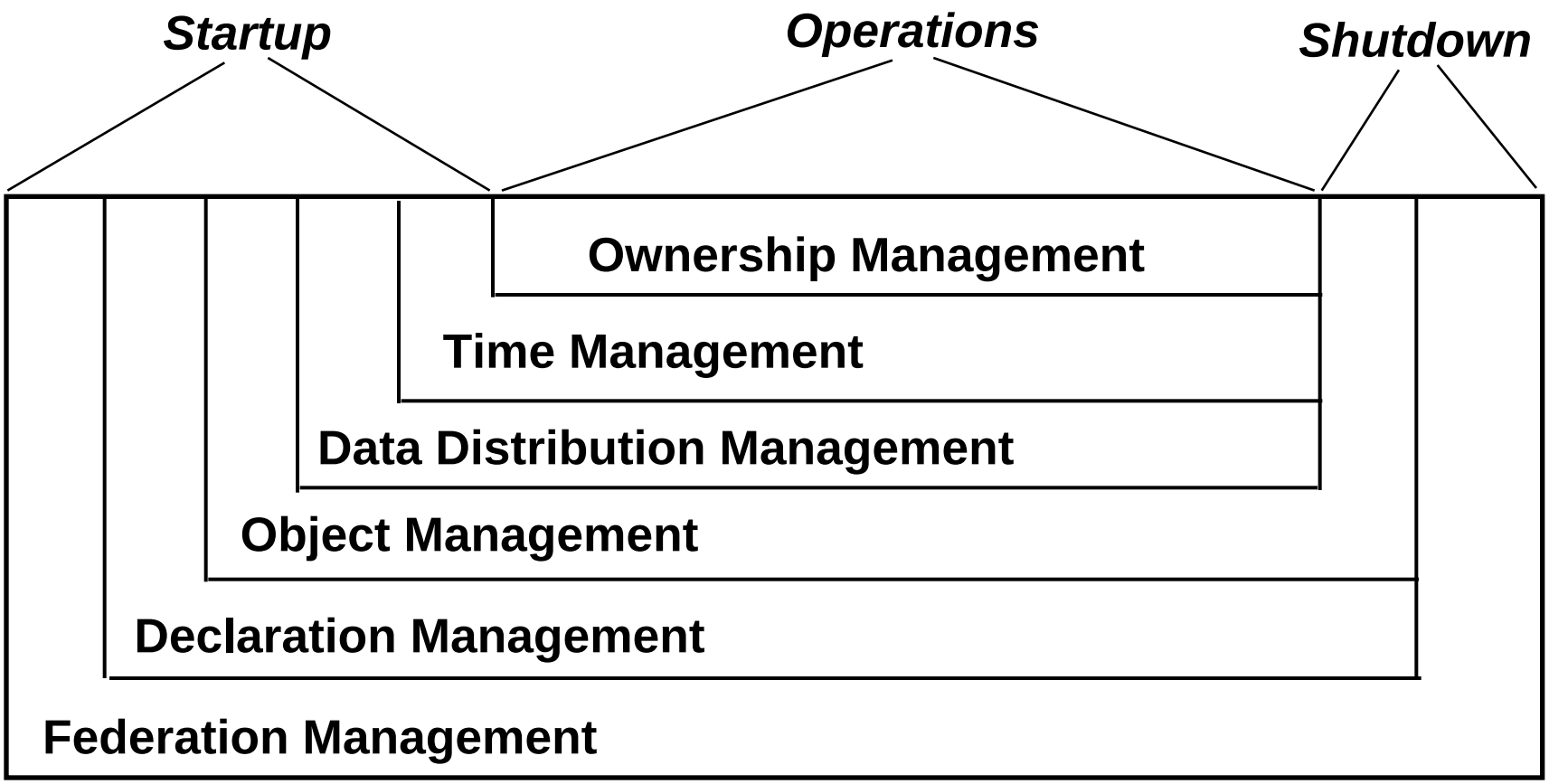
II. HLA I/F SPECIFICATION SERVICE GROUPS

III. USE OF THE INTERFACE IN A TYPICAL FEDERATION EXECUTION

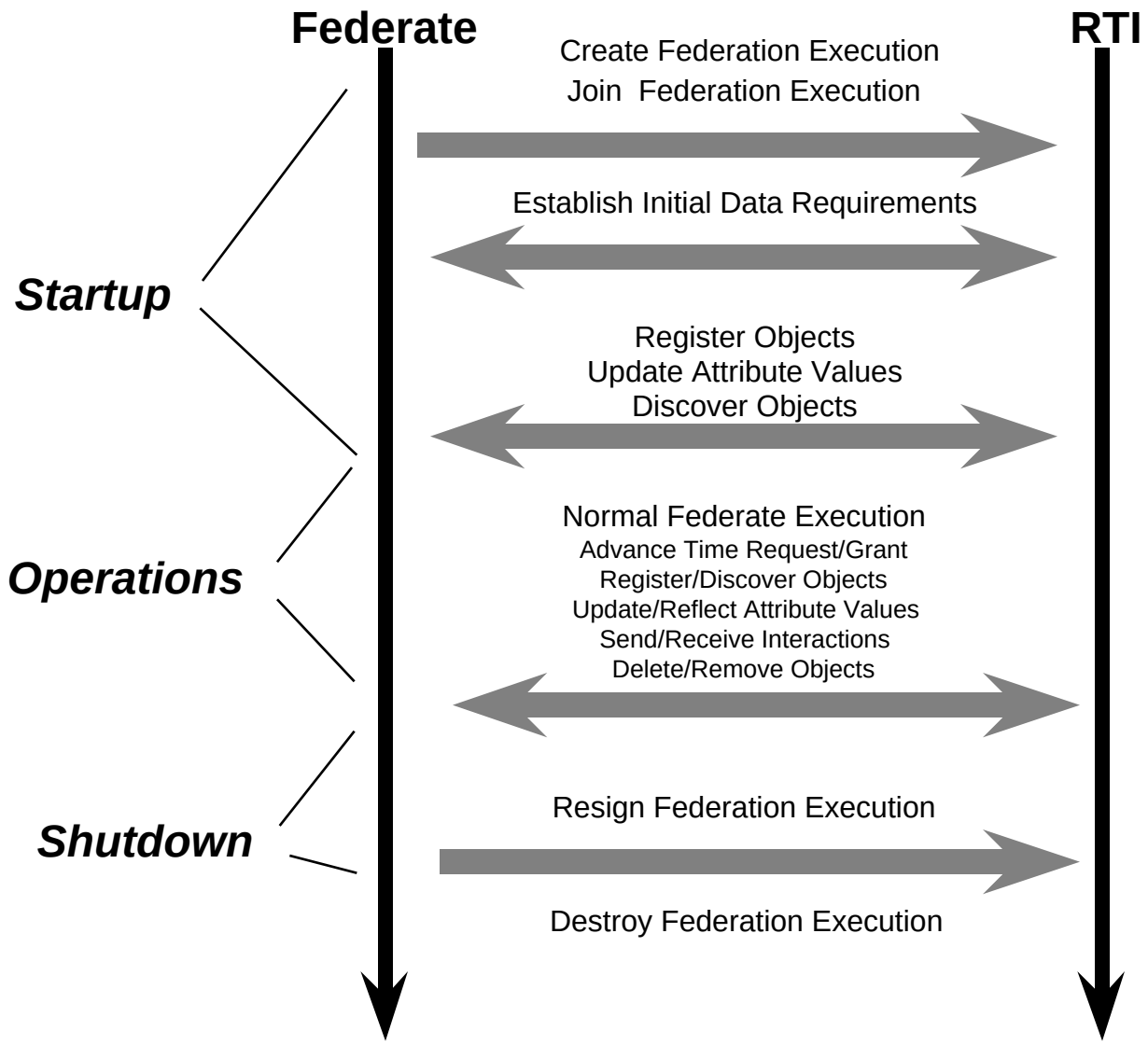
IV. SUPPORTING DOCUMENTS AND TECHNICAL PAPERS

V. CONCLUSION

Use of Interface Over Life Cycle of a Typical Federation Execution



Overview of a Typical Federation Execution Life Cycle



Startup:
Federation Management

Federate manages a federation execution during startup...

Create Federation Execution [2.1]

Initializes the RTI with Federation Specific Information

- **Establish class, attribute, interaction names and hierarchies, as defined in FOM**
- **Set default vales for ordering and transport services**
- **Establish names and dimensions of routing spaces (if using data distribution management services)**

Join Federation Execution [2.3]

RTI affiliates a federate with the federation execution

Startup:
Declaration Management

Federate states desire to generate and receive data...

Publish Object Class and Publish Interaction Class [3.1, 3.2]

Federate declares desire to update attributes of classes of objects or send classes of interactions

Subscribe Object Class Attribute & Subscribe Interaction Class [3.3, 3.4]

Federate informs RTI of desire to discover object attributes and classes of interactions

Startup:
Object Management

Federate creates, discovers, and modifies objects ...

Request ID [4.1]

Federate requests unique ID numbers from RTI

Register Object [4.2]

Federate links an object ID with an instance of an object

Update Attribute Values [4.3]

Federate provides current attribute values of an instance of an object

Discover Object [4.4]

RTI notifies federate of an object's existence

Reflect Attribute Values [4.5]

RTI informs federates of the updated values of objects

Startup:
Data Distribution Management

Federate makes declarations during startup for management of data distribution...

Create Update Region [7.1]

Federate identifies region of interest within a routing space for sending updates and interactions

Create Subscription Region [7.2]

Federate identifies region of interest within a routing space for receiving updates and interactions

Associate Update Region [7.3]

Federate associates an update region of interest with an object

Startup:
Time Management

Federate using event ordering sets initial transportation and ordering categories as well as lookahead...

- **Note the need to set default transportation and event ordering requirements for object classes and interactions in RID**

Set Lookahead [6.5]

Federate sets initial lookahead window if conservatively synchronized desiring causal behavior

Operations:
Object Management

Federates update and receive simulation events as determined by Declaration and DDM services...

Update Attributes [4.3]

Federate provides current values for attributes of simulation objects being modeled

Send Interactions [4.6]

Federate provides interaction data for simulation object's actions toward other objects

Reflect Attributes [4.5]

RTI delivers simulation object attribute updates

Receive Interactions [4.7]

RTI delivers interaction data to subscribing federates

Operations:
Time Management

Federates request message delivery and advancement along the federation time axis. RTI grants permission to advance...

Time Advance Request , Next Event Request [6.7 - 6.8]

Federate requests time advancement along with associated event delivery of TSO messages for constrained federates

Flush Queue Request [6.9]

Federates request event delivery and time advancement

Time Advance Grant [6.10]

RTI honors time advancement requests

Request .... Time [6.1, 6.3, 6.4]

Federate queries RTI for time values

Operations:
Declaration Management

Federate states or modifies desire to generate and receive data. RTI notifies federate of the need to send data when and if needed...

Publish Object Class , Interaction Class [3.1 - 3.2]

Federate declares desire to generate data

Subscribe Object Class Attribute , Interaction Class [3.3 - 3.4]

Federate declares desire to receive data

Control Updates , Interactions [3.5 - 3.6]

RTI notifies Federate when and if data is needed

Operations:
Data Distribution Management

Federate modifies data routing by changing bounds on regions, associating an object with a new region or creating/deleting regions...

Create Update , Subscription Region [7.1 - 7.2]

Federate creates regions of interest for generating and receiving data

Associate Update Region [7.3]

Federate associates an object with an update region

Change Thresholds [7.4]

RTI notifies federate to change boundaries on region of interest

Modify Region [7.5]

Federate modifies boundaries on region of interest

Operations:
Federation Management

Federate may request or be requested to pause / resume or save / restore state ...

Request Pause, Initiate Pause, Pause Achieved [2.5-2.7]

Federates and RTI coordinate federation execution pause

Request Resume, Initiate Resume, Resume Achieved [2.8-2.10]

Federates and RTI coordinate federation execution resumption

Request Federation Save, Initiate Federation Save, Federation Save Achieved [2.12-2.14]

Federates and RTI coordinate federation execution state save

Request Restore, Initiate Restore, Restore Achieved [2.15-2.17]

Federates and RTI coordinate federation restoration

Operations:
Ownership Management

Federate may request or be requested to divest or acquire ownership of attributes based on Publish declarations...

Request Attribute Ownership Divestiture [5.1]

Federate requests divestiture of owned object attribute

Request Attribute Ownership Assumption [5.2]

RTI asks federate to acquire object attribute ownership

Attribute Ownership Acquisition Notification [5.4]

RTI notification that federate now owns object attribute

Attribute Ownership Divestiture Notification [5.3]

RTI notification that federate no longer owns object attribute

Shutdown:
Federation Management

All federates resign at the end of execution, and the federation is destroyed .

Resign Federation Execution [2.4]

Federate notifies RTI that it no longer desires to participate in the federation execution

Destroy Federation Execution [2.2]

Federate notifies RTI that the federation execution is no longer desired

Outline

I. INTRODUCTION

II. HLA I/F SPECIFICATION SERVICE GROUPS

III. USE OF THE INTERFACE IN A TYPICAL FEDERATION EXECUTION

IV. SUPPORTING DOCUMENTS AND TECHNICAL PAPERS

V. CONCLUSION

HLA Technical Library

- DMSO has established an online “public library” for the M&S community, available through the DMSO Web page <http://www.dmso.mil>
- Contents related to Interface Specification:
  - HLA Baseline Definition (Rules, Interface Specification, Object Model Template)
  - HLA Glossary
  - Interface Specification Supporting Documents (Test Procedures, Time Management, APIs, Data Distribution Management)
  - HLA Compliance Checklist
  - HLA Security Architecture
  - Additional briefings and documents

On-Line Documentation

- **Proceedings and products of the AMG appear under the subtopic “Common Technical Framework for M&S”, under “High Level Architecture”. DMSO home page site is:**

  http://www.dmso.mil/

- **Specific questions can be directly addressed to DMSO via electronic mail at:**

  hla@msis.dmso.mil

Interface Specification Documents

- **Defining Documents**
  - **HLA Interface Specification Version 1.1, 4 February 1997, available at http://www.dmso.mil**
- **Supporting Documents**
  - **Application Programming Interfaces for the HLA Runtime Infrastructure, John Cosby, et. al, Spring 97 SIW**
  - **Understanding the HLA Interface, D. Clark, P. Hoare, 15th DIS Workshop, paper # 96-15-028**
  - **An Introduction to the HLA Interface Specification and Object Model Template Test Procedures, M. Loper, D. Roberts, 15th DIS Workshop, paper # 96-15-098**
  - **The HLA Interface Specification and Application Programmers Interface (API), T. Stark, R. Weatherly, A. Wilson, 14th DIS Workshop, paper # 96-14-122**
  - **...**

Outline

I. INTRODUCTION

II. HLA I/F SPECIFICATION SERVICE GROUPS

III. USE OF THE INTERFACE IN A TYPICAL FEDERATION EXECUTION

IV. SUPPORTING DOCUMENTS AND TECHNICAL PAPERS

V. CONCLUSION

Conclusion

- **Evolutionary Status**
  - Interface Specification V1.0 baselined 15 August 1996
  - Interface Specification V1.1 baselined 4 February 1997
  - Subsequent versions TBA
- **Standing**
  - Monotonically convergent
  - Partially validated (contingent on RTI Devel.) in multiple venues
- **Bottom Line**
  - Design goals are being achieved
  - Use of Interface Specification to support Federation and RTI development has been demonstrated
  - Use of the Interface Specification to facilitate interoperability has been demonstrated...facilitation of reuse is being explored
  - I/F Specification is ‘safe to use’ for HLA system design and development